Common Vulnerability Scoring System (CVSS) Version 2

Karen Scarfone, NIST

Acknowledgements

- FIRST conference presentation, Gavin Reid, Cisco Systems
- CVSS v2 Complete Documentation, FIRST CVSS-SIG

Disclaimer: Certain commercial equipment or materials are identified in this presentation in order to adequately specify and describe the use of CVSS. Such identification is not intended to imply recommendation or endorsement by NIST, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

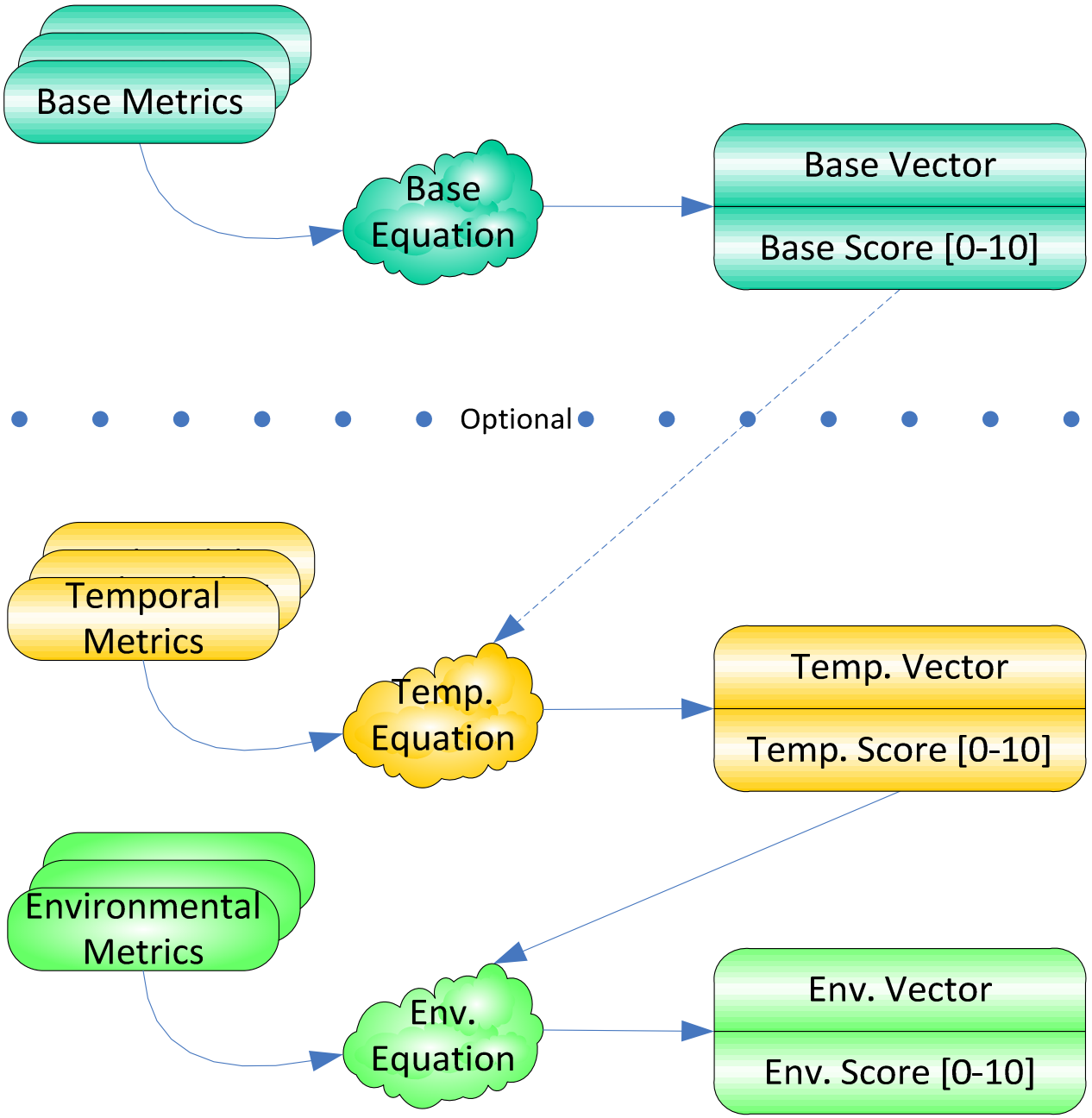
Agenda

- Introduction and overview of CVSS
- Why CVSS?
- Base scores
- Temporal scores
- Environmental scores
- Example
- Score usage

Overview

- Common Vulnerability Scoring System (CVSS)
- A universal way to convey vulnerability severity and help determine urgency and priority of responses
  - 20+ new vulnerabilities a day for organizations to prioritize and mitigate
- A set of metrics and formulas
- Solves problem of incompatible scoring systems
- Under the custodial care of FIRST CVSS-SIG
- Open, usable, and understandable by anyone
- Version 2 released in June 2007, adopted by SCAP

Metrics and Scores



Base Metric Group

- Most fundamental qualities of a vulnerability
- Does not change; intrinsic and immutable
- Represents general vulnerability severity
- Two subsets of three metrics each:
  - Exploitability: Access Vector, Access Complexity, Authentication
  - Impact: Confidentiality, Integrity, Availability

Access Vector (AV)

- Measures how remote an attacker can be to exploit a vulnerability
- Local (L): The vulnerability is only exploitable locally (physical access or local account)
- Adjacent Network (A): The attacker must have access to either the broadcast or collision domain of the vulnerable software
- Network (N): The vulnerable software is bound to the network stack and the attacker does not need local or adjacent network access to exploit it

Access Complexity (AC)

- Measures the complexity of attack required to exploit the vulnerability once an attacker has access to the target system
- High (H): Specialized access conditions exist, such as the attacker already having elevated privileges, spoofing additional systems, or relying on obvious and convoluted social engineering methods
- Medium (M): The access conditions are somewhat specialized, such as only certain systems or users being able to perform attacks, the affected configuration being uncommon, or some information gathering being required
- Low (L): Specialized access conditions or extenuating circumstances do not exist, such as the affected product typically requiring access to a wide range of systems and users, the affected configuration being the default, and the attack requiring little skill or information gathering

Authentication (Au)

- Measures the number of times an attacker must authenticate to a target *once the system has been accessed* in order to exploit a vulnerability
- Multiple (M): Exploiting the vulnerability requires that the attacker authenticate two or more times (e.g., first OS, then application), even if the same credentials are used each time
- Single (S): One instance of authentication is required
- None (N): Authentication is not required to exploit the vulnerability

Confidentiality Impact (C)

- Measures the impact on confidentiality of a successfully exploited vulnerability
- None (N): No impact on confidentiality
- Partial (P): Considerable informational disclosure, such as access to some files or certain database tables
- Complete (C) : Total information disclosure; the attacker can read all of the system's data (including files and memory)

Integrity Impact (I)

- Measures the impact to integrity of a successfully exploited vulnerability
- None (N): No impact on integrity
- Partial (P): Modification of some system files or information
- Complete (C): Total compromise of system integrity; the attacker can modify any data (files, memory, etc.) on the target system

Availability Impact (A)

- Measures the impact to availability of a successfully exploited vulnerability
- None (N): No impact on availability
- Partial (P): Reduced performance or interruptions in resource availability
- Complete (C): Total shutdown of the affected resource

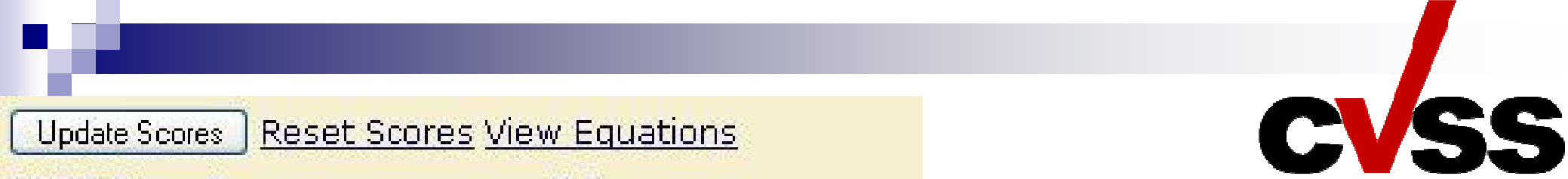
Base Scoring

- Computed by vendors and coordinators
- Each metric has a number assigned to each possible value
  - AccessComplexity: high = 0.35, medium = 0.61, low = 0.71
  - Integrity: none = 0.0, partial = 0.275, complete = 0.66
- The metrics' values are combined with formulas that give different weights to the base metrics
- Base subscores for impact and exploitability
- The final base score is between 0.0 and 10.0
  - 60% of impact subscore + 40% of exploitability subscore

Base Vector

- A vector is a representation of the values assigned to the CVSS metrics
- Every CVSS score should be accompanied by the corresponding vector, so that people can see the components of the score and validate them
- CVSS base vector has the following form:

  (AV:[L,A,N]/AC:[H,M,L]/Au:[M,S,N]/C:[N,P,C]/I:[N,P,C]/A:[N,P,C])
- Sample vector:
  (AV:N/AC:L/Au:N/C:P/I:P/A:P)



Update Scores | Reset Scores | View Equations

CVSS Base Score	7.5
Impact Subscore	6.4
Exploitability Subscore	10
CVSS Temporal Score	Undefined
CVSS Environmental Score	Undefined
Overall CVSS Score	**7.5**

Base Score Metrics

Exploitability Metrics

AccessVector	Network
AccessComplexity	Low
Authentication	None

Impact Metrics

ConfImpact	Partial
IntegImpact	Partial
AvailImpact	Partial

NVD CVSS Calculator

Temporal Metric Group

- Time-dependent qualities of a vulnerability
- Represents urgency at a specific point in time
- Optional—only the base metrics are mandatory
- Three temporal metrics:
  - Exploitability
  - Remediation Level
  - Report Confidence

Exploitability (E)

- Measures the current state of exploit techniques or code availability
- Unproven (U): No exploit code is available
- Proof-of-Concept (POC): Proof-of-concept exploit code or an impractical exploit is available
- Functional (F) : Functional exploit code is available
- High (H): Either there is functional mobile autonomous code or no exploit is required (manual trigger) and details are widely available
- Not Defined (ND): No value assigned—skip this metric in calculating the score

Remediation Level (RL)

- Measures the level of available remediation solutions
- Official Fix (OF): Complete vendor solution available, such as an official patch or upgrade
- Temporary Fix (TF): Official temporary fix available
- Workaround (W): Unofficial non-vendor solution available
- Unavailable (U): Either no solution available or it is impossible to apply
- Not Defined (ND): No value assigned—skip this metric in calculating the score

Report Confidence (RC)

- Measures the degree of confidence in the existence of the vulnerability and the credibility of reports
- Unconfirmed (UC): A single unconfirmed source or possibly multiple conflicting reports; little confidence
- Uncorroborated (UR): Multiple non-official sources, possibly including independent security companies or research organizations
- Confirmed (C): Vendor has reported/confirmed a problem with its own product, or functional exploit code is available
- Not Defined (ND): No value assigned—skip this metric in calculating the score

Temporal Scoring

- Computed by vendors and coordinators
- Designed to be re-evaluated at specific intervals as a vulnerability ages
- Each metric has a number assigned to each possible value
- The temporal formula starts with the base score and alters it based on the temporal metrics' values
- Temporal vector has the following form:

  (E:[U,POC,F,H,ND]/RL:[OF,TF,W,U,ND]/ RC:[UC,UR,C,ND])

Environmental Metric Group

- Qualities of a vulnerability specific to a particular IT environment
- Optional—only the base metrics are mandatory
- Five environmental metrics
  - Collateral Damage Potential
  - Target Distribution
  - Security Requirements
    - Confidentiality requirement
    - Integrity requirement
    - Availability requirement

Collateral Damage Potential (CDP)

- Measures the potential for loss of life or physical assets through damage or theft of property or equipment, and economic loss of productivity or revenue
- None (N): No potential for physical assets, productivity or revenue damage
- Low (L): Slight damage or loss of revenue or productivity
- Low-Medium (LM): Moderate damage or loss of revenue or productivity
- Medium-High (MH): Significant damage or loss of revenue or productivity
- High (H): Catastrophic damage or loss of revenue or productivity
- Not Defined (ND): No value assigned—skip this metric in calculating the score
- Each organization has to define precisely what “slight”, “moderate”, “significant”, and “catastrophic” mean

Target Distribution (TD)

- Measures the proportion of vulnerable systems in an environment
- None (N): No target systems exist, or targets are highly specialized and exist only in a laboratory setting (0%)
- Low (L): Targets exist on a small scale (1-25%)
- Medium (M): Targets exist on a medium scale (26-75%)
- High (H): Targets exist on a considerable scale (76-100%)
- Not Defined (ND): No value assigned—skip this metric in calculating the score

Security Requirements

- Customize score based on the importance of the targets to the organization in terms of the targets' confidentiality, integrity, and availability
- Confidentiality requirement (CR), integrity requirement (IR), availability requirement (AR): each affects the weight of the corresponding base metric (C, I, A)
- Effect on the organization or associated individuals:
  - Low (L): Likely to have only a limited adverse effect
  - Medium (M): Likely to have a serious adverse effect
  - High (H): Likely to have a catastrophic adverse effect
  - Not Defined (ND): No value assigned—skip this metric in calculating the score

Environmental Scoring

- Computed by end users
- Each metric has a number assigned to each possible value
- Formula starts with the temporal score and alters it based on the environmental metrics' values
- User organizations can use this to prioritize responses within their own environments
- Environmental vector has the following form:

  (CDP:[N,L,LM,MH,H,ND]/TD:[N,L,M,H,ND]/
  CR:[L,M,H,ND]/IR:[L,M,H,ND]/AR:[L,M,H,ND])

Example - **CVE-2003-0062**

- Consider CVE-2003-0062: Buffer Overflow in NOD32 Antivirus. In February 2003, a buffer overflow vulnerability was discovered in Linux and Unix versions prior to 1.013 that could allow local users to execute arbitrary code with the privileges of the user executing NOD32. To trigger the buffer overflow, the attacker creates a directory with an overly long name that will be scanned by the antivirus software.

Example (cont.)

- Since the vulnerability is exploitable only to a user locally logged into the system, the Access Vector is "Local".
- The Access Complexity is "Low" because an attacker with local access can create a directory with an overly long name at will, and it is very likely that the attacker can do so in a location that will be scanned by the antivirus software.
- Authentication is set to "None" because the attacker does not need to authenticate to any additional system.

Example (cont.)

- If an administrative user were to run the virus scan, causing the buffer overflow, then a full system compromise would be possible. Since the most harmful case must be considered, each of the three Impact metrics is set to “Complete”.
- Together, these metrics produce a base score of 7.2
  - Impact subscore 10.0, exploitability subscore 3.9
- The base vector for this vulnerability is AV:L/AC:L/Au:N/C:C/I:C/A:C.

Example (cont.)

- Partial exploit code has been released, so the Exploitability metric is set to "Proof-Of-Concept". The vendor has released updated software, giving a Remediation Level of "Official-Fix" and Report Confidence of "Confirmed". These three metrics adjust the base score to give a temporal score of 5.6.
- Assuming that confidentiality, integrity, and availability are roughly equally important for the targeted systems, and depending on the values for Collateral Damage Potential and Target Distribution, the environmental score could vary between 0.0 ("None", "None") and 7.8 ("High", "High").

Score Usage

- Intended as a generalization, primarily for comparing the relative severity of different vulnerabilities
  - Does not reflect the full likelihood of attack (such as a popular product being targeted more often than a rarely used product)
  - Does not take into account whether deployed security controls may prevent exploits
  - May be errors or minor inaccuracies in scoring
- National Vulnerability Database assigns rankings according to CVSS base scores
  - Low: 0.0 to 3.9
  - Medium: 4.0 to 6.9
  - High: 7.0 to 10.0

Future Work

- Identifying possible changes to CVSS v2 and its documentation to address inaccuracies and ambiguities
  - Studying CVSS v2 scores from NVD
  - Discussing differences in scoring among analysts and analyst organizations
- Revamping temporal and environmental metrics
- Applying CVSS to other types of scoring, such as for security-related software configuration settings (Common Configuration Scoring System, CCSS)

Questions?

- karen.scarfone@nist.gov

Links

- CVSS-SIG: http://www.first.org/cvss/
- CVSS v2 Complete Documentation: http://www.first.org/cvss/cvss-guide.html
- NIST Interagency Report 7435: http://csrc.nist.gov/publications/nistir/
- NIST NVD: http://nvd.nist.gov/cvss.cfm?version=2 and http://nvd.nist.gov/cvss.cfm?calculator&version=2